We are bug fixers

Things you already know:
- How to use Scratch to place and join blocks of code together to achieve a goal.
- Understand what an algorithm is.

Knowledge you will gain:
You will test existing coding to explore the bugs that exist. You will become proficient in identifying bugs that prevent code from running and also performance related bugs that would enhance the overall coding.

- Knowledge and understand of **performance and quality bugs**: Bugs that make code run slowly or are incomplete.
- Knowledge and understand of **multithread bugs**: Bugs caused by two or more processes running at the same time when delays and timing are required.
- Knowledge and understand of **conceptual bugs**: Bugs that are built in to the code by design due to a misconception or lack of thought around a problem.
- Knowledge and understand of **arithmetical bugs**: Bugs caused with maths issues, so for example a division program might handle complete numbers, but may not be able to cope with remainders or decimals.

Vocabulary

Debug: To correct a program that doesn't work or to refine a program to make it perform better.

Logical order: The computer follows the instructions one after another, starting at the top of your code and working its way down to the bottom.

Variables: A value that can be changed such as speed or distance.

Specific skills/understanding
Debugging skills including logical systematic checking through blocks of coding. Understanding that all variables can be changed to have an impact on the outcome of the code.

Ongoing skill set
Navigating files and folders to access shortcuts to coding. Improve speed of logging in to the network.